Un choix de dix thèmes étudiés cette année pour revoir les principes de l'algorithmique.

1. Suites : limite infinie
2. Suites : suite convergente
3. Suites : somme des termes
4. Équations : résolution par balayage
5. Équations : résolution par dichotomie
6. Calcul intégral : méthode des rectangles
7. Calcul intégral : méthode de Monte-Carlo
8. Probabilités : simulation d'une expérience aléatoire
9. Probabilités conditionnelles : évolution d'un processus
10. Lois continues : approximation par la loi normale

Limite infinie d'une suite

On étudie la suite définie pour tout $n \geq 0$ par

$$\boxed{u_n = \frac{n}{2} + (-1)^n}$$

Il s'agit d'examiner son comportement lorsque n devient grand et plus précisément d'illustrer la définition suivante du cours :

on dit que la suite (u_n) tend vers $+\infty$ lorsque pour tout réel K il existe un rang n à partir duquel les termes de la suite sont supérieurs ou égaux à K

Dans ce but on considère l'algorithme décrit par le pseudocode suivant ;

```
VARIABLES
   n, u, p sont des nombres
   N est un nombre
DÉBUT
   p prend la valeur 1
   N prend la valeur 9
   POUR n allant de 0 à N
      u prend la valeur n/2 + p
      afficher u
      p prend la valeur -1 * p
   FIN de la boucle POUR
FIN
```

Questions

1. Faire fonctionner l'algorithme sur une feuille de papier de façon à obtenir les 10 premiers termes de la suite (u_n) . Pour cela on peut reproduire et compléter le tableau suivant :

n		0	1	2	etc...	9
u	_					
p	1					
N	9					

2. Taper et tester l'algorithme avec Algobox : vérifier que les valeurs affichées sont les mêmes que celles qui sont dans le tableau de la question précédente.
Quelle modification faut-il apporter pour afficher les 20 premiers termes de la suite ?

3. On veut maintenant adapter l'algorithme de façon à afficher le premier entier n tel que $u_n \geq 500$

a) Pourquoi une boucle POUR n'est pas adaptée à ce genre de question ?

b) Modifier l'algorithme précédent en remplaçant la boucle POUR par une boucle TANT_QUE
- la condition devra être : `u < 500`
- la variable N n'a plus aucune utilité

c) Taper et tester l'algorithme modidié.

4. Démontrer que pour tout $n \geq 1002$ on a bien $u_n \geq 500$
(le résultat est admis pour un réel K quelconque)

Suite convergente

On étudie la suite définie pour tout $n \geq 0$ par

$$u_0 = 28 \qquad \text{et} \qquad u_{n+1} = \frac{3 - u_n}{5}$$

Il s'agit d'examiner son comportement lorsque n devient grand et plus précisément d'illustrer la définition suivante du cours :

on dit que la suite (u_n) tend vers le réel ℓ lorsque tout intervalle ouvert contenant ℓ contient tous les termes de la suite à partir d'un certain rang

Dans ce but on considère l'algorithme décrit par le pseudocode suivant ;

```
VARIABLES
   n, u sont des nombres
   N est un nombre
DÉBUT
   u prend la valeur 28
   N prend la valeur ......
   POUR n allant de 1 à N
      u prend la valeur ...............
      afficher u
   FIN de la boucle POUR
FIN
```

Questions

1. Compléter le pseudocode de façon que l'algorithme affiche les 15 premiers termes de la suite.

2. Taper et tester l'algorithme avec Algobox.
Quelle conjecture peut-on faire à propos de la limite éventuelle de la suite ?

3. On veut maintenant adapter l'algorithme de façon à afficher le premier entier n tel que u_n soit dans l'intervalle $]0,4\,;0,6[$: on doit remplacer la boucle POUR par une boucle TANT_QUE .

a) La condition « *u est dans l'intervalle* $]0,4\,;0,6[$ » se traduit par : `0.4 < u ET u < 0,6`
Quel est son contraire logique ?

b) Modifier l'algorithme précédent en utilisant une boucle TANT_QUE

c) Taper et tester l'algorithme modidié.

4. Déterminer le premier rang n_0 pour lequel $u_n \in]0,49\,;0,51[$
Démontrer ensuite que si $u_n \in]0,49\,;0,51[$, alors $u_{n+1} \in]0,49\,;0,51[$
Quel résultat obtient-on pour la suite (u_n) ?

5. Le résultat précédent est admis dans le cas général : tout intervalle ouvert contenant 0,5 va contenir tous les termes de la suite à partir d'un certain rang.

Somme des termes d'une suite

Le Lièvre est à la poursuite de la Tortue.



La tortue a encore 5 m d'avance mais elle s'est endormie.
Voyant cela, le lièvre ralentit son allure. La première seconde, il fait un bond de 1m, la seconde suivante il fait un bond de 0,50 m et ainsi de suite : chaque seconde, il fait un bond moitié moindre que le bond précédent.
Au bout de combien de secondes le lièvre aura-t-il rattrapé la tortue ?

Mise en forme du problème.

On note u_n la distance en mètres que le lièvre a parcouru au bout de n secondes.
Puisque la première seconde le lièvre bondit de 1m , on aura : $u_1 = 1$
Sachant que la seconde suivante, il fait un bond de 0,50m, on aura : $u_2 = 1 + 0,50 = 1,50$
Et en général, on pose $\boxed{u_n = 1 + 0,50 + 0,25 + \text{ etc}\dots}$ (la somme comprenant n termes)

L'algorithme suivant permet de calculer la distance parcourue au bout de n secondes.

```
VARIABLES
   i et b sont des nombres
   n et u sont des nombres
DÉBUT
   u prend la valeur 0
   b prend la valeur 1
   n prend la valeur ......
   POUR i allant de 1 à n
      u prend la valeur ..............
      b prend la valeur ..............
   FIN de la boucle POUR
   afficher u
FIN
```

Questions

1. Compléter le pseudocode de façon que l'algorithme affiche la distance parcourue au bout de 5 secondes en tenant compte des informations suivantes :
  ⋆ `n` est le nombre de secondes au bout desquelles on calcule la distance parcourue
  ⋆ `i` est un compteur
  ⋆ `u` est la distance parcourue au bout de `n` secondes
  ⋆ `b` est le bond fait par le lièvre : il est donc divisé par 2 chaque seconde

2. Taper et tester l'algorithme avec Algobox.

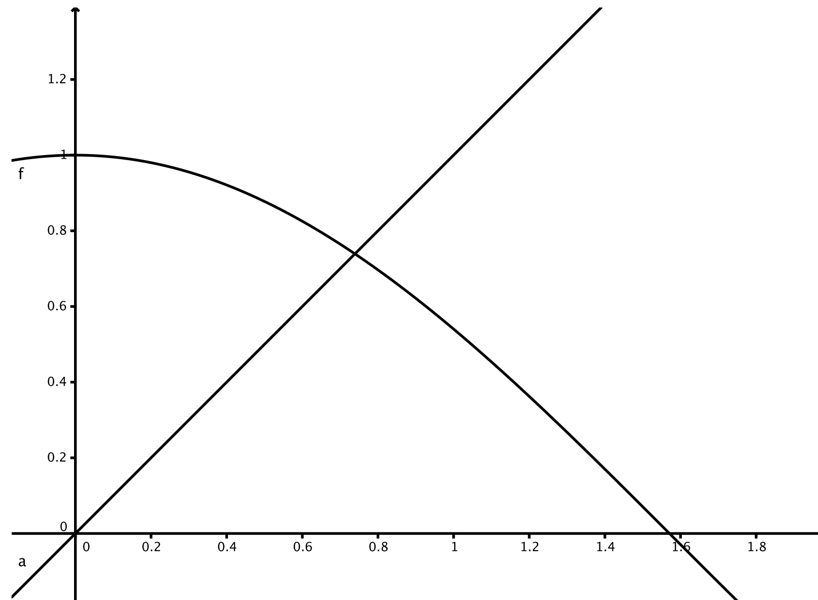
3. Modifier l'algorithme de façon que l'utilisateur puisse rentrer le nombre de secondes au bout desquelles il veut connaître la distance parcourue par le lièvre.

4. Proposer une réponse à la question initiale en testant plusieurs valeurs de n

5. Démontrer la conjecture précédente
(on pourra utiliser le résultat du cours sur la somme des termes d'une suite géométrique pour obtenir un majorant de u_n et comparer ce nombre avec 5)

Résolution approchée d'une équation par balayage

Figure.



On a représenté ci-dessus la courbe d'équation $y = \cos x$ et la droite d'équation $y = x$.
On veut déterminer l'abscisse du point d'intersection des deux courbes qui est visible sur cette figure.
On peut observer graphiquement que cette abscisse est comprise entre 0 et 1.

Mise en forme du problème.

On note $f(x)$ la différence entre les deux fonctions représentées :

$$f(x) = x - \cos x$$

Il est alors équivalent de résoudre l'équation $\boxed{f(x) = 0}$

Ne sachant pas déterminer une solution exacte, on se contente d'un encadrement de celle-ci.

L'algorithme suivant teste si la fonction f change de signe entre deux abscisses x_1 et x_2 de l'intervalle $[0\ ;1]$. Une série de 10 tests est effectuée.

```
VARIABLES
   a , b , p , x1 , x2 , y1 et y2 sont des nombres
   i est un nombre
DÉBUT
   a prend la valeur 0
   b prend la valeur 1
   p prend la valeur (b - a) / 10
   x1 prend la valeur a
   x2 prend la valeur x1 + p
   POUR i allant de 1 à 10
      y1 prend la valeur x1 - cos(x1)
      y2 prend la valeur ...............
      si y1*y2 < 0 afficher x1 et x2
      x1 prend la valeur x1 + p
      x2 prend la valeur ...............
   FIN de la boucle POUR
FIN
```

Questions

1. Calculer $f'(x)$ et démontrer que cette dérivée est strictement positive sur $[0\,;1]$
   Démontrer que l'équation $f(x) = 0$ a une solution α et une seule sur cet intervalle.
2. Compléter le pseudocode de façon que l'algorithme affiche les bornes de l'intervalle qui contient α en tenant compte des informations suivantes :
   - ⋆ `p` est le pas dont on augmente x_1 et x_2 à chaque itération
   - ⋆ `i` est un compteur
3. Taper et tester l'algorithme avec Algobox. Indiquer l'encadrement de α qui est obtenu.
4. On veut améliorer la précision de la réponse.
   Pour cela, on va exécuter trois fois la même boucle POUR comme le suggère le pseudocode suivant :

```
VARIABLES
   a , b , p , x1 , x2 , y1 et y2 sont des nombres
   i et j sont des nombres
DÉBUT
   a prend la valeur 0
   b prend la valeur 1
   POUR j allant de 1 à 3
      p prend la valeur ...............
      x1 prend la valeur ...............
      x2 prend la valeur ...............
      POUR i allant de 1 à 10
         y1 prend la valeur ...............
         y2 prend la valeur ...............

         SI y1*y2 < 0 ALORS
            afficher x1 et x2
            a prend la valeur x1
            b prend la valeur x2
         FIN de l'instruction SI

         x1 prend la valeur x1 + p
         x2 prend la valeur ...............
      FIN de la boucle POUR
   FIN de la boucle POUR
FIN
```

5. Compléter et tester l'algorithme proposé.
6. Comment l'adapter de façon que l'utilisateur puisse donner au départ la précision souhaitée ?

Résolution approchée d'une équation par dichotomie

On sait que la fonction exponentielle est continue et strictement croissante et que ses limites en $-\infty$ et en $+\infty$ sont respectivement 0 et $+\infty$. Par conséquent l'équation

$$\boxed{e^x = 2}$$

a une solution α et une seule qu'on se propose d'encadrer par « *dichotomie* » .

Il est immédiat de vérifier que α est entre 0 et 1 : on a donc au départ un encadrement d'amplitude 1 . On va tester le milieu de cet intervalle, c'est-à-dire 0, 5 . Il y a deux possiblités :

- si $e^{0,5} > 2$ alors α est entre 0 et 0,5
- si $e^{0,5} < 2$ alors α est entre 0,5 et 1

On obtient ainsi un encadrement d'amplitude 0,5.

L'algorithme suivant applique cette méthode jusqu' à obtenir une précision suffisante :

```
VARIABLES
   x1 , x2 , xm , y1 , y2 et ym sont des nombres
   p est un nombre
DÉBUT
   x1 prend la valeur 0
   x2 prend la valeur 1
   p prend la valeur x2 - x1
   TANT_QUE p > 0.1
      xm prend la valeur (x1 + x2) / 2
      ym prend la valeur exp(xm)
      si ym < 2 alors x1 prend la valeur xm
      sinon ..............
      p prend la valeur ..............
   FIN de la boucle TANT_QUE
   ............
FIN
```

Questions

1. Compléter l'algorithme de façon qu'il affiche les bornes d'un intervalle d'amplitude inférieure à 0,1.

2. Taper et tester l'algorithme avec Algobox. Indiquer l'encadrement de α qui est obtenu.

3. Comment améliorer la précision du résultat ?

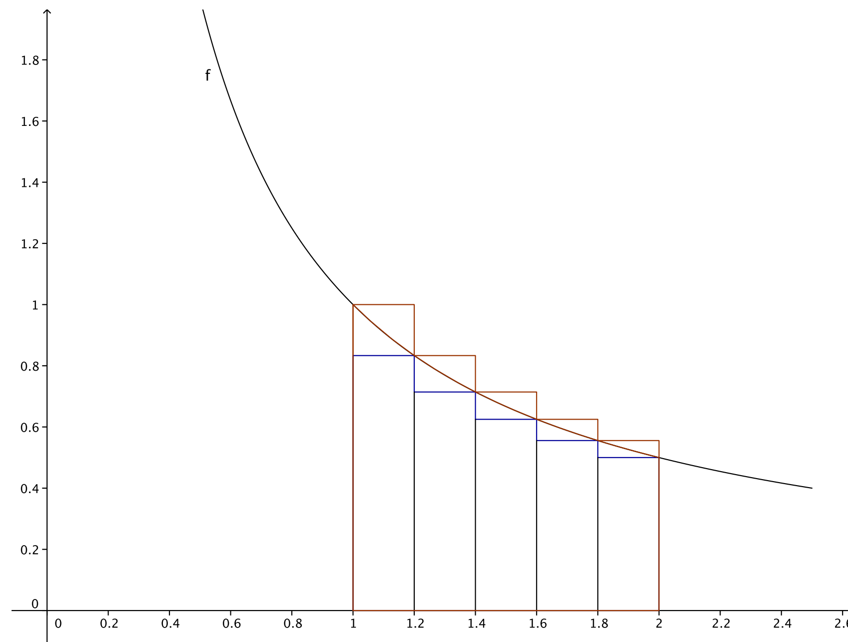
4. Comment l'adapter de façon que l'utilisateur puisse donner au départ la précision souhaitée ?

5. On considère l'équation $-x^3 - 5x + 12 = 0$.
Démontrer que cette équation a une solution et une seule dans $\mathbb{R}$.
Adapter l'algorithme précédent de façon à obtenir un encadrement de cette solution d'amplitude inférieure à 10^{-5} .
Combien d'itérations de la boucle TANT_QUE ont été nécessaires ?

Approximation d'une aire par la méthode des rectangles

On souhaite calculer l'aire sous la courbe de la fonction inverse entre 1 et 2 .

Figure.



L'intervalle [1 ; 2] a été partagé en 5 petits intervalles d'amplitude 0,2 .
Le domaine dont on veut calculer l'aire est contenu dans la réunion de 5 petits rectangles dits « *à gauche* » de hauteurs respectives $f(1)$, $f(1,2)$, $f(1,4)$, $f(1,6)$ et $f(1,8)$.
Ce même domaine contient la réunion de 5 petits rectangles dits « *à droite* » de hauteurs respectives $f(1,2)$, $f(1,4)$, $f(1,6)$, $f(1,8)$ et $f(2)$.

Questions

1. Déterminer avec la calculatrice l'aire de la réunion des rectangles « *à gauche* » puis l'aire de la réunion des rectangles « *à droite* » .
En déduire un encadrement de l'aire du domaine. Comparer avec la valeur exacte.

2. On veut améliorer la précision du résultat.
On note n le nombre d'intervalles construits entre 1 et 2 et h le *pas* utilisé .
Chaque rectangle a pour base un intervalle $[a\,;b]$ dont les valeurs successives sont $[1\,;1,2]$, $[1,2\,;1,4]$ etc $\ldots[1,8\,;2]$.
Dans un premier temps, on considère l'algorithme suivant

```
i , n , h , a , b , G et D sont des nombres
n prend la valeur 5
h prend la valeur 1/n
a prend la valeur 1
b prend la valeur a + h
POUR i allant de 1 à n
   G prend la valeur .........
   D prend la valeur .........
   a prend la valeur a + h
   b prend la valeur a + h
FIN de POUR
afficher G
afficher D
```

Compléter cet algorithme de telle façon qu'il affiche en sortie la somme des aires des rectangles à gauche et la somme des aires des rectangles à droite.

3. On souhaite maintenant que l'utilisateur indique en entrée le nombre n d'intervalles à utiliser.
Proposer et tester un algorithme qui réponde à cette question.

4. Une assez bonne approximation de l'aire est obtenue en faisant la moyenne des deux approximations précédentes (méthode des *trapèzes*) : modifier et tester l'algorithme précédent de façon à implémenter cette méthode.

5. Déterminer une valeur approchée de $\displaystyle\int_0^1 2t\ e^{-t^2}\ \mathrm{d}t$. Quelle est la valeur exacte ?

---

1. Aire des rectangles à gauche : $G = [f(1) + f(1,2) + f(1,4) + f(1,6) + f(1,8)] \times 0,2 \approx 0,746$
Aire des rectangles à droite : $D = [f(1,2) + f(1,4) + f(1,6) + f(1,8) + f(2)] \times 0,2 \approx 0,646$
Encadrement de l'aire sous la courbe : $0,646 \leq A \leq 0,746$
Valeur exacte : $A = \displaystyle\int_1^2 \frac{\mathrm{d}x}{x} = [\ln x]_1^2 = \ln 2 \approx 0,693$

2. Algorithme permettant de calculer l'aire des rectangles à gauche et l'aire des rectangles à droite.

```
i , n , h , a , b , G et D sont des nombres
n prend la valeur 5
h prend la valeur 1/n
a prend la valeur 1
b prend la valeur a + h
POUR i allant de 1 à n
   G prend la valeur G + h / a
   D prend la valeur D + h / b
   a prend la valeur a + h
   b prend la valeur b + h
FIN de POUR
afficher G
afficher D
```

3. Algorithme avec choix du nombre d'intervalles.

```
i , n , h , a , b , G et D sont des nombres
lire n
etc .........
```

4. Implémentation de la méthode des trapèzes.

```
i , n , h , a , b , T , G et D sont des nombres
lire n
h prend la valeur 1/n
a prend la valeur 1
b prend la valeur a + h
POUR i allant de 1 à n
   etc .........
FIN de POUR
T prend la valeur (G + D) / 2
Afficher T
```

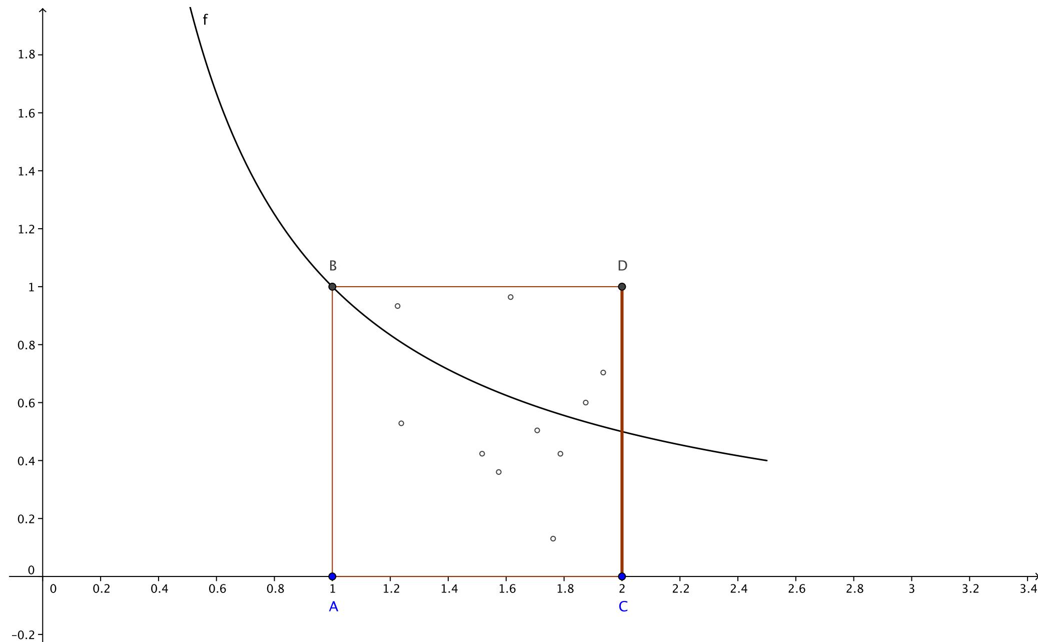
Résultats numériques :

n	5	10	50
G	0.74563492	0.7187714	0.69817218
D	0.64563492	0.6687714	0.68817218
T	0.69563492	0.6937714	0.69317218

Approximation d'une aire par la méthode de Monte Carlo

On souhaite de nouveau calculer l'aire sous la courbe de la fonction inverse entre 1 et 2 .
Ce domaine est contenu dans un carré ABDC que l'on a représenté ci-dessous.

Figure.



On choisit 10 points au hasard dans le carré ABDC : parmi ces dix points 6 se trouvent en-dessous de la courbe, alors que 4 points se trouvent au-dessus.
La probabilité de choisir un point du carré sous la courbe est approximativement égale à 0,6 .
Or cette probabilité est le rapport entre l'aire sous la courbe et l'aire du carré : une estimation de l'aire sous la courbe est donc 0,6 .

Questions

1. On veut améliorer la précision du résultat.
On note n le nombre de points choisis au hasard dans le carré ABDC .
Un point M de coordonnées (x, y) est en-dessous de la courbe lorsque $y \leq \frac{1}{x}$
Dans un premier temps, on considère l'algorithme suivant où T désigne le nombre de points en-dessous de la courbe :

```
i , n , x , y , T sont des nombres
n prend la valeur 100
POUR i allant de 1 à n
   x prend la valeur 1 + random ()
   y prend la valeur random ()
   SI y <= 1 / x ALORS ...............
FIN de POUR
afficher ...............
```

Compléter cet algorithme de telle façon qu'il affiche en sortie une estimation de l'aire sous la courbe.

2. On souhaite maintenant que l'utilisateur indique en entrée le nombre n de points à choisir.
Proposer et tester un algorithme qui réponde à cette question.
Pour quelle valeur de n obtient-on expérimentalement une erreur inférieure à 10^{-1} ?

3. Déterminer une valeur approchée de π par la méthode de Monte Carlo.

1. Première estimation de l'aire .

```
i , n , x , y , T sont des nombres
n prend la valeur 100
POUR i allant de 1 à n
   x prend la valeur 1 + random ()
   y prend la valeur random ()
   SI y <= 1 / x ALORS T prend la valeur T + 1
FIN de POUR
afficher T / 100
```

2. Estimation avec choix du nombre de points .

```
i , n , x , y , T sont des nombres
lire n
etc .........
```

Résultats numériques.

n	10	100	1000	10 000
T / n	0,8	0,68	0,692	0,6983

On peut estimer que si $n \geq 100$, l'erreur faite en approchant l'aire par cette méthode est inférieure à 10^{-1} .

3. Il nous faut un domaine dont l'aire soit en rapport avec le nombre π :

- le disque de centre O et de rayon 1 a pour aire π et pour équation $x^2 + y^2 \leq 1$
- si on impose $y \geq 0$, on obtient un demi-disque de centre O et de rayon 1 qui a pour aire $\pi/2$
- si on impose $x \geq 0$ et $y \geq 0$, on obtient un quart-de-disque d'aire $\pi/4$

La méthode de Monte Carlo est facile à adapter au cas du quart-de disque : on choisit au hasard des points dans le carré $0 \leq x \leq 1$ et $0 \leq y \leq 1$ et on teste pour chaque point si $x^2 + y^2 \leq 1$.

```
i , n , x , y , T sont des nombres
lire n
POUR i allant de 1 à n
   x prend la valeur random ()
   y prend la valeur random ()
   SI x*x + y*y <= 1 ALORS T prend la valeur T + 1
FIN de POUR
afficher 4 * T / n
```

Résultats numériques.

n	10	100	1000	10 000
4 * T / n	2,4	3,08	3,172	3,1272

Problème du Duc de Toscane

On lance trois dés bien équilibrés.
Est-il plus probable que la somme des chiffres obtenus soit égale à 9 ou à 10 ?

Questions

1. Montrer qu'il y a autant de façons de décomposer 9 ou 10 en une somme de trois entiers compris entre 1 et 6 , ces entiers étant pris dans un ordre quelconque.
2. L'algorithme suivant simule l'expérience consistant à lancer trois dés et à calculer leur somme.
```
i , n , x , y , z , s , f et g sont des nombres
n prend la valeur 1000
POUR i allant de 1 à n
   x prend la valeur 1 + floor(6 * random ())
   y prend la valeur 1 + floor(6 * random ())
   z prend la valeur 1 + floor(6 * random ())
   s prend la valeur x + y + z
   SI s = .........ALORS ...............
FIN de POUR
f prend la valeur g / n
afficher ..............
```
   Compléter l'algorithme de façon qu'on obtienne une approximation de la probabilité de l'événement « *la somme est égale à 9* »
3. Comparer le résultat précédent avec la probabilité de l'événement « *la somme est égale à 10* »
4. Est-il plus probable d'obtenir une somme égale à 9 ou à 10 ? Justifier votre réponse.

1. Décompositions de 9 en une somme de trois entiers compris entre 1 et 6 :

```
1 + 2 + 6
1 + 3 + 5
1 + 4 + 4
2 + 2 + 5
2 + 3 + 4
3 + 3 + 3
```

Décompositions de 10 en une somme de trois entiers compris entre 1 et 6 :

```
1 + 3 + 6
1 + 4 + 5
2 + 2 + 6
2 + 3 + 5
2 + 4 + 4
3 + 3 + 4
```

2. Approximation de la probabilité de l'événement « *la somme est égale à 9* » .

```
i , n , x , y , z , s , f et g sont des nombres
n prend la valeur 1000
POUR i allant de 1 à n
   x prend la valeur 1 + floor(6 * random ())
   y prend la valeur 1 + floor(6 * random ())
   z prend la valeur 1 + floor(6 * random ())
   s prend la valeur x + y + z
   SI s = 9 ALORS g prend la valeur g + 1
FIN de POUR
f prend la valeur g / n
afficher f
```

Résultats numériques.

n	10	100	1000	10 000
g / n	0,1	0,2	0,122	0,1176

3. Probabilité de l'événement « *la somme est égale à 10* »

n	10	100	1000	10 000
g / n	0,2	0,08	0,134	0,118

4. Est-il plus probable d'obtenir une somme égale à 9 ou à 10 ? Difficile de conclure avec les données expérimentales.
Il y a $6^3 = 216$ issues toutes équiprobables.
Une décomposition en trois entiers distincts est réalisée par 6 triplets.
Si une décomposition comprend deux fois le même nombre, elle n'est plus réalisée que par 3 triplets.
Enfin, si une décomposition comprend trois fois le même nombre, elle est réalisée par un seul triplet.
La somme 9 est donc réalisée par $3 \times 6 + 2 \times 3 + 1 \times 1 = 25$ triplets .
Alors que la somme 10 est réalisée par $3 \times 6 + 3 \times 3 = 27$ triplets.
Il est donc plus probable que la somme soit 10.

Le marché du téléphone

Dans une ville imaginaire, le marché des télécommunications est partagé entre deux compagnies A et B. Initialement, la compagnie A truste 80% des abonnements. Mais la compagnie B est très agressive et cherche à augmenter sa part de marché.
Dans cette ville, la règle veut que chaque habitant doit à la fin de chaque année confirmer son abonnement auprès de la compagnie qu'elle a choisie ou au contraire changer de compagnie.
Une étude statistique conduit aux résultats suivants :

- il y a 6 chances sur 10 qu'à la fin d'une année une personne cliente de A souscrive un abonnement auprès de B
- il y a 3 chances sur 10 qu'à la fin d'une année une personne cliente de B souscrive un abonnement auprès de A

Quelle compagnie aura à long terme la plus grosse part de marché dans cette ville ?

Questions

On note a_n la proportion d'habitants qui sont clients de la compagnie A après n années.
De même b_n désigne la proportion d'habitants qui sont clients de la compagnie B après n années.
On a donc au départ $a_0 = 0,8$ et $b_0 = 0,2$.

1. L'algorithme suivant détermine l'évolution de la part de marché de chaque compagnie.

```
i , n , a , b , A , B sont des nombres
a prend la valeur .........
b prend la valeur .........
n prend la valeur .........
POUR i allant de 1 à n
   A prend la valeur .......................
   B prend la valeur .......................
   afficher n
   afficher A
   afficher B
   a prend la valeur A
   b prend la valeur B
FIN de la boucle POUR
```

2. Compléter et tester l'algorithme. Quelle conjecture peut être proposée ?

3. **a)** Démontrer que $a_{n+1} = 0,4a_n + 0,3b_n$

b) En déduire $a_{n+1} = 0,1a_n + 0,3$

c) Démontrer que $u_n = a_n - \dfrac{1}{3}$ est géométrique

d) En déduire la limite de (u_n) puis de (a_n).

4. Quelle réponse peut-on apporter à la question initiale ?

1. D'après le texte :

$$\begin{cases} a_{n+1} & = & 0,4\,a_n + 0,3\,b_n \\ b_{n+1} & = & 0,6\,a_n + 0,7\,b_n \end{cases}$$

Évolution de la part de marché de chaque compagnie.

```
i , n , a , b , A , B sont des nombres
a prend la valeur 0,8
b prend la valeur 0,2
n prend la valeur 10
POUR i allant de 1 à n
   A prend la valeur 0.4 * a + 0.3 * b
   B prend la valeur 0.6 * a + 0.7 * b
   afficher n
   afficher A
   afficher B
   a prend la valeur A
   b prend la valeur B
FIN de la boucle POUR
```

2. Résultats numériques.

n	0	1	2	3	4	5
a_n	0,8	0,38	0,338	0,334	0,333	0,333
b_n	0,2	0,62	0,662	0,666	0,667	0,667

Conjecture : à terme les compagnies A et B auront respectivement $\dfrac{1}{3}$ et $\dfrac{2}{3}$ du marché.

3. a) Les clients de A pendant l'année $n+1$ sont constitués de 40% des clients de A pendant l'année n et de 30% des clients de B pendant l'année n.

b) On remplace b_n par $1 - a_n$: $a_{n+1} = 0,4\,a_n + 0,3\,(1 - a_n) = 0,1\,a_n + 0,3$

c) La suite auxiliaire $u_n = a_n - \dfrac{1}{3}$ est géométrique :

$$\begin{aligned} u_{n+1} &= a_{n+1} - \frac{1}{3} \\ &= \frac{a_n}{10} + \frac{3}{10} - \frac{1}{3} \\ &= \frac{a_n}{10} - \frac{1}{30} \\ &= \frac{1}{10}\left(a_n - \frac{1}{10}\right) \\ &= 0,1\,u_n \end{aligned}$$

Sachant $u_0 = \dfrac{8}{10} - \dfrac{1}{3} = \dfrac{7}{15}$, on peut écrire $u_n = \dfrac{7}{15}\left(\dfrac{1}{10}\right)^n$.

On en déduit : $a_n = \dfrac{7}{15}\left(\dfrac{1}{10}\right)^n + \dfrac{1}{3}$

d) Puisque $-1 < \dfrac{1}{10} < 1$, on a $\lim \left(\dfrac{1}{10}\right)^n = 0$.

On en déduit $\lim u_n = 0$ puis $\lim a_n = \dfrac{1}{3}$.

4. La conjecture précédente est vérifiée.

Approximation d'une loi binomiale par la loi normale

Dans cet exercice, n désigne un entier naturel strictement positif.
Au départ, on suppose $n = 10$ et on étudie l'expérience aléatoire suivante :

- une pièce de monnaie est lancée n fois de suite
- on compte le nombre de fois où on a obtenu « *pile* » au cours des n lancers

Cette expérience aléatoire peut-être simulée par l'algorithme suivant :

```
j , P et X sont des nombres
X prend la valeur 0
POUR j allant de 1 à 10
   P prend la valeur floor(2 *random())
   SI P = 0 ALORS X prend la valeur X + 1
FIN de la boucle POUR
Afficher X
```

Dans cet algorithme, le lancer de la pièce est simulé par le choix d'un nombre entier aléatoire égal à 0 (*pile*) ou à 1 (*face*).
On note désormais X la variable aléatoire égale au nombre de « *pile* » au cours d'une série de n lancers.

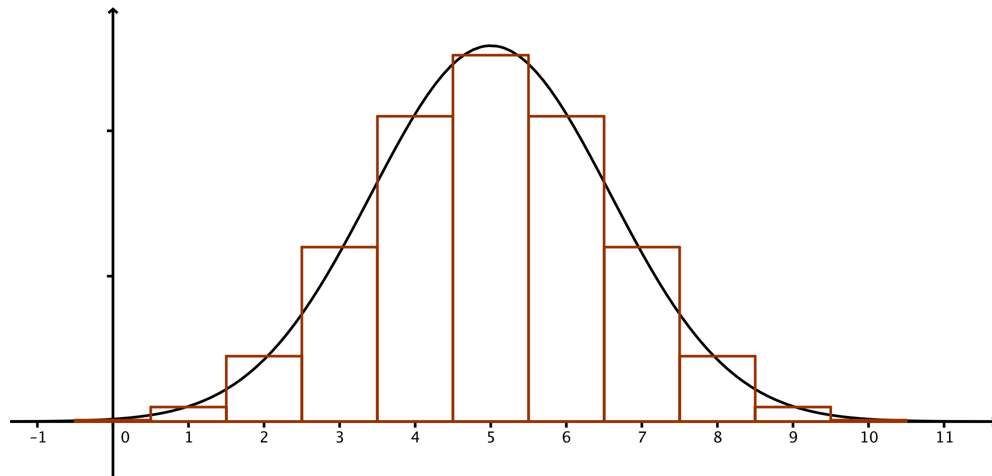
Questions

1. Quelle loi de probabilité est suivie par X ?
En déduire l'espérance μ, la variance V et l'écart-type σ de X.

2. On veut déterminer expérimentalement $p(X = 5)$ en simulant un grand nombre de fois l'expérience initiale. Proposer et tester un algorithme qui réponde à la question.
On pourra utiliser les variables suivantes :
`N` : nombre de simulations (déterminé en entrée par l'utilisateur)
`i` : compteur associé aux simulations (ce compteur est distinct du compteur `j`)
`G` : nombre de cas où X prend la valeur 5
`F` : fréquence associée à l'événement $(X = 5)$ qui doit être affichée en sortie
Indiquer les valeurs obtenues dans un tableau et comparer avec la valeur théorique.

3. Montrer que l'événement $(\mu - \sigma \leq X \leq \mu + \sigma)$ est la réunion des événements $(X = 4)$, $(X = 5)$ et $(X = 6)$.
Adapter l'algorithme précédent pour obtenir une approximation de $p_1 = p(\mu - \sigma \leq X \leq \mu + \sigma)$.
Tester et comparer avec la valeur théorique.

4. Dans cette question, on va estimer p_1 par une méthode différente.



Sur cette figure chaque rectangle a une aire égale à $p(X = k)$ où k est l'abscisse indiquée en-dessous du rectangle.

On a également représenté une fonction f qui « *approche* » le dessin constitué par ces rectangles : cette fonction s'appelle « *la densité de probabilité de la loi normale d'espérance* $\mu = 5$ *et de variance* $\sigma^2 = 2,5$ ».
On démontre qu'une très bonne approximation de p_1 est l'aire sous la courbe de f entre les abscisses $a = \mu - \sigma$ et $b = \mu + \sigma$. On va déterminer l'aire de ce domaine avec différents outils.
Dans chacun des cas considérés, les variables a , b , μ et σ devront être remplacées par leurs valeurs numériques.

a) Lancer **Geogebra** et définir dans la barre de saisie la fonction densité :

`f(x) = Normale[`μ `,` σ `, x]`

Déterminer l'aire sous la courbe entre les bornes indiquées :

`IntégraleDomaine[f , 0 , a , b]`

b) Lancer **Algobox** et définir la fonction numérique :

`F1(x) = ALGOBOX_LOI_NORMALE(`μ `,` σ `, x)`

Cette fonction calcule l'aire sous la courbe de f qui est située à gauche de l'abscisse x.
Il suffit donc dans un algorithme de calculer la probabilité recherchée en écrivant l'instruction :

`p1 prend la valeur F1(b) - F1(a)`

c) Lancer un **tableur** et utiliser la fonction `LOI.NORMALE`
Cette fonction renvoie la valeur de la densité en `x` avec :

`= LOI.NORMALE(x ;` μ `;` σ`)`

La même fonction renvoie l'aire sous la courbe de f qui est située à gauche de l'abscisse x avec :

`= LOI.NORMALE(x ;` μ `;` σ `; 1)`

Il suffit donc de taper dans une cellule du tableur :

`= LOI.NORMALE(b ;` μ `;` σ `; 1) - LOI.NORMALE(a ;` μ `;` σ `; 1)`

d) Vérifier les résultats obtenus avec votre **calculatrice**
TI : utiliser la fonction `normalcdf(a , b ,` μ `,` σ`)`
CASIO : dans le menu `STAT DIST NORM Ncd` préciser les valeurs de `Lower` , `Upper` , σ et μ

5. Application. On lance 100 fois une pièce de monnaie et on note X le nombre de « *pile* ».
En convenant de noter μ l'espérance de X et σ son écart-type, déterminer la probabilité des événements suivants :

$(X \leq 40)$
$(X \geq 90)$
$(\mu - \sigma \leq X \leq \mu + \sigma)$
$(\mu - 2\sigma \leq X \leq \mu + 2\sigma)$
$(\mu - 3\sigma \leq X \leq \mu + 3\sigma)$

---

1. X suit une loi binomiale de paramètres $n = 10$ et $p = 0,5$
L'espérance est $\mu = n\,p = 5$, la variance est $V = n\,p\,(1-p) = 2,5$ et l'écart-type est $\sigma = \sqrt{2,5} \approx 1,58$

2. Estimation de $p(X = 5)$

```
i , j , P , F , G , N et X sont des nombres
lire N
POUR i allant de 1 à N
   X prend la valeur 0
   POUR j allant de 1 à 10
      P prend la valeur floor(2 *random())
      SI P = 0 ALORS X prend la valeur X + 1
   FIN de la boucle POUR
   SI X = 5 ALORS G prend la valeur G + 1
FIN de la boucle POUR
F prend la valeur G / N
Afficher F
```

Résultats numériques pour l'événement $(X = 5)$.

N	10	100	1 000	10 000
F	0,3	0,231	0,239	0,2495

Remarque : la valeur théorique est $p(X = 5) = \binom{10}{5} 0,5^{10} \approx 0,246$

3. On vérifie $\mu - \sigma \approx 3,42$ et $\mu + \sigma \approx 6,58$. Par conséquent :

$$\mu - \sigma \leq X \leq \mu + \sigma \iff X = 4 \quad \text{ou} \quad X = 5 \quad \text{ou} \quad X = 6$$

Algorithme permettant d'estimer $p_1 = p(E - \sigma \leq X \leq E + \sigma)$: il suffit de remplacer dans l'algorithme précédent le test

```
X = 5
```

par le test :

```
4 <= X ET X <= 6
```

Résultats numériques pour p_1 :

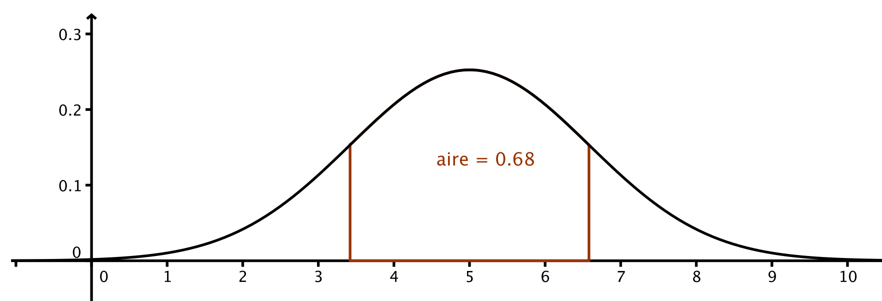
N	10	100	1 000	10 000
F	0,5	0,61	0,668	0,6574

Calcul de la valeur théorique :

$$\begin{aligned} p_1 &= p(X = 4) + p(X = 5) + p(X = 6) \\ &= \binom{10}{4} 0,5^{10} + \binom{10}{5} 0,5^{10} + \binom{10}{6} 0,5^{10} \\ &= 672 \cdot 0,5^{10} \approx 0,656 \end{aligned}$$

4. Approximation par la loi normale.

a) **Geogebra**



b) Algobox

```
...
p prend la valeur F1(5 + 1.58) - F1(5 - 1.58)
afficher p
...
```

`Fonction numérique utilisée : F1(x) = ALGOBOX_LOI_NORMALE(5 , 1.58 , x)`

c) Tableur : OpenOffice

```
= LOI.NORMALE(6,58 ; 5 ; 1,58 ; 1) - LOI.NORMALE(3,42 ; 5 ; 1,58 ; 1)
```

d) Calculatrices

TI : `normalcdf(3.42 , 6.58 , 5 , 1.58)`
CASIO : renseigner `Lower Upper` σ μ dans le menu `Normal C.D.` et calculer

5. Application au cas de 100 lancers. Dans ce cas :
l'espérance est $\mu = 100 \times 0,5 = 50$
la variance est $V = 100 \times 0,5 \times 0,5 = 25$
l'écart-type est $\sigma = \sqrt{25} = 5$
Les résultats suivants ont été obtenus avec un tableur :
$p(X \leq 40) = 0,02275$
$p(X \geq 90) = 0$
$p(45 \leq X \leq 55) = 0,68269$
$p(40 \leq X \leq 60) = 0,95450$
$p(35 \leq X \leq 65) = 0,99730$
